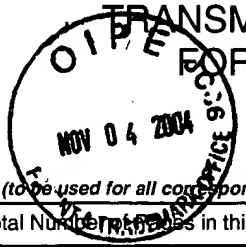
PTO/SB/21 (09-04)
Approved for use through 07/31/2006. OMB 0651-0031
U.S. Patent and Trademark Office: U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

AF
IFW

TRANSMITTAL FORM



(to be used for all correspondence after initial filing)

Total Number of Pages in this Submission: 35

Application Number	09/967,031
Filing Date	September 27, 2001
First Named Inventor	Lokpraveen B. MOSUR et al.
Art Unit	2187
Examiner Name	Kimberly N. McLEAN-MAYO
Attorney Docket Number	Intel 2207/11305

ENCLOSURES *(check all that apply)*

- ☒ Fee Transmittal Form
- ☐ Fee Attached
- ☐ Amendment / Reply
- ☐ After Final
- ☐ Affidavits/declaration(s)
- ☐ Extension of Time Request
- ☐ Express Abandonment Request
- ☐ Information Disclosure Statement
- ☐ Certified Copy of Priority Document(s)
- ☐ Reply to Missing Parts/ Incomplete Application
- ☐ Reply to Missing Parts under 37 CFR1.52 or 1.53
- ☐ Drawing(s)
- ☐ Licensing-related Papers
- ☐ Petition
- ☐ Petition to Convert to a Provisional Application
- ☐ Power of Attorney, Revocation Change of Correspondence Address
- ☐ Terminal Disclaimer
- ☐ Request for Refund
- ☐ CD, Number of CD(s)
- ☐ Landscape Table on CD
- ☐ After Allowance Communication to TC
- ☒ Appeal Communication to Board of Appeals and Interferences
- ☒ Appeal Communication to TC (Appeal Notice, Brief, Reply Brief)
- ☐ Proprietary Information
- ☐ Status Letter
- ☒ Other Enclosure(s) *(please identify below)*:
  1. **Postcard**

Remarks

SIGNATURE OF APPLICANT, ATTORNEY, OR AGENT

Firm	Kenyon & Kenyon		
Signature	Stephen T. Neal		
Printed Name	Stephen T. Neal		
Date	November 1, 2004	Reg. No.	47,815

CERTIFICATE OF TRANSMISSION/MAILING

I hereby certify that this correspondence is being facsimile transmitted to the USPTO or deposited with the United States Postal Service with sufficient postage as first class mail in an envelope addressed to: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450 on the date shown below. **M/S: APPEAL BRIEFS – PATENTS**

Signature	Pilar Rodriguez		
Typed or printed name	Pilar Rodriguez	Date	November 1, 2004

PTO/SB/17 (10-04v2)
Approved for use through 07/31/2006. OMB 0651-0032
U.S. Patent and Trademark Office: U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

FEE TRANSMITTAL for FY 2005

Effective 10/01/2004. Patent fees are subject to annual revision.

☐ Applicant claims small entity status. See 37 CFR 1.27

TOTAL AMOUNT OF PAYMENT ($) **340.00**

Complete if Known	
Application Number	09/967,031
Filing Date	September 27, 2001
First Named Inventor	Lokpraveen B. MOSUR et al.
Examiner Name	Kimberly N. McLEAN-MAYO
Art Unit	2187
Attorney Docket No.	Intel 2207/11305

METHOD OF PAYMENT *(check all that apply)*

☐ Check ☐ Credit card ☐ Money Order ☐ Other ☐ None

☒ Deposit Account:

Deposit Account Number: **11-0600**

Deposit Account Name: **Kenyon & Kenyon**

The Director is authorized to: *(check all that apply)*
☒ Charge fee(s) indicated below ☒ Credit any overpayments
☒ Charge any additional fee(s) or any underpayment of fee(s)
☒ Charge fee(s) indicated below, **except for the filing fee** to the above-identified deposit account.

FEE CALCULATION

1. BASIC FILING FEE

Large Entity Fee Code	Large Entity Fee ($)	Small Entity Fee Code	Small Entity Fee ($)	Fee Description	Fee Paid
1001	790	2001	395	Utility filing fee	
1002	350	2002	175	Design filing fee	
1003	550	2003	275	Plant filing fee	
1004	790	2004	395	Reissue filing fee	
1005	160	2005	80	Provisional filling fee	
				SUBTOTAL (1)	($) 0

2. EXTRA CLAIM FEES FOR UTILITY AND REISSUE

			Extra Claims		Fee from below		Fee Paid
Total Claims		-20 **	=	X	18.00	=	
Independent Claims		-3 **	=	X	88.00	=	
Multiple Dependent				X		=	

Large Entity Fee Code	Large Entity Fee ($)	Small Entity Fee Code	Small Entity Fee ($)	Fee Description
1202	18	2202	9	Claims in excess of 20
1201	88	2201	44	Independent claims in excess of 3
1203	300	2203	150	Multiple dependent claim, if not paid
1204	88	2204	44	** Reissue independent claims over original patent
1205	18	2205	9	** Reissue claims in excess of 20 and over original patent

SUBTOTAL (2) ($) 0

***or number previously paid, if greater; For Reissues, see above*

FEE CALCULATION (continued)

3. ADDITIONAL FEES

Large Entity Fee Code	Large Entity Fee ($)	Small Entity Fee Code	Small Entity Fee ($)	Fee Description	Fee Paid
1051	130	2051	65	Surcharge - late filing fee or oath	
1052	50	2052	25	Surcharge - late provisional filing fee or cover sheet.	
1053	130	1053	130	Non-English specification	
1812	2,520	1812	2,520	For filing a request for *ex parte* reexamination	
1804	920*	1804	920*	Requesting publication of SIR prior to Examiner action	
1805	1,840*	1805	1,840*	Requesting publication of SIR after Examiner action	
1251	110	2251	55	Extension for reply within first month	
1252	430	2252	215	Extension for reply within second month	
1253	980	2253	490	Extension for reply within third month	
1254	1,530	2254	765	Extension for reply within fourth month	
1255	2,080	2255	1,040	Extension for reply within fifth month	
1401	340	2401	170	Notice of Appeal	
1402	340	2402	170	Filing a brief in support of an appeal	340.00
1403	300	2403	150	Request for oral hearing	
1451	1,510	1451	1,510	Petition to institute a public use proceeding	
1452	110	2452	55	Petition to revive – unavoidable	
1453	1,370	2453	685	Petition to revive – unintentional	
1501	1,370	2501	685	Utility issue fee (or reissue)	
1502	490	2502	245	Design issue fee	
1503	660	2503	330	Plant issue fee	
1460	130	1460	130	Petitions to the Commissioner	
1807	50	1807	50	Processing fee under 37 CFR 1.17 (q)	
1806	180	1806	180	Submission of Information Disclosure Stmt	
8021	40	8021	40	Recording each patent assignment per property (times number of properties)	
1809	790	2809	395	Filing a submission after final rejection (37 CFR § 1.129(a))	
1810	790	2810	395	For each additional invention to be examined (37 CFR § 1.129(b))	
1801	790	2801	395	Request for Continued Examination (RCE)	
1802	900	1802	900	Request for expedited examination of a design application	
				Other fee (specify)	

*Reduced by Basic Filing Fee Paid

SUBTOTAL (3) ($) 340.00

SUBMITTED BY					Complete *(if applicable)*
Name (Print/Type)	Stephen T. Neal	*Registration No. (Attorney/Agent)*	47,815	*Telephone*	(408) 975-7500
Signature	Stephen T. Neal			*Date*	November 1, 2004

WARNING: Information on this form may become public. Credit card information should not be included on this form. Provide credit card information and authorization on PTO-2038.

Patent

Attorney Docket No.: Intel 2207/11305
U.S. Serial No.: 09/967,031
Assignee: Intel Corporation

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

APPELLANTS	:	Lokpraveen B. MOSUR et al.
SERIAL NO.	:	09/967,031
FILED	:	September 27, 2001
FOR	:	LIST BASED METHOD AND APPARATUS FOR SELECTIVE AND RAPID CACHE FLUSHES
GROUP ART UNIT	:	2187
EXAMINER	:	Kimberly N. McLEAN-MAYO

M/S: APPEAL BRIEFS - PATENTS
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Certificate of Mailing

I hereby certify that this correspondence is being deposited with the United States Postal Service as First Class Mail in an envelope addressed to: M/S: APPEAL BRIEFS – PATENTS, Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450, Attention: Board of Patent Appeals and Interferences, on

Dated: November 1, 2004 Signature Pilar Rodriguez

Pilar Rodriguez

ATTENTION: Board of Patent Appeals and Interferences

APPELLANT'S BRIEF

Dear Sir:

This brief is in furtherance of the Notice of Appeal, filed in this case on August 26, 2004, and received in the USPTO on August 30, 2004.

11/05/2004 MMEKONEN 00000007 110600 09967031
01 FC:1401 340.00 DA

11/05/2004 MMEKONEN 00000019 110600 09967031
01 FC:1402 340.00 DA

1. REAL PARTY IN INTEREST

The real party in interest in this matter is Intel Corporation. (Recorded January 23, 2002, Reel/Frame 012537/0223).

2. RELATED APPEALS AND INTERFERENCES

There are no related appeals.

3. STATUS OF THE CLAIMS

Claims 1-30 are pending in this application. Claims 1-30 were rejected under 35 U.S.C. §103(a). This appeal is an appeal from the rejection of claims 1-30.

4. STATUS OF AMENDMENTS

No claims were amended in the previous response.

5. SUMMARY OF THE INVENTION

An apparatus and a method for rapidly flushing a cache memory device, including a list structure to track changes in a cache, which may be implemented on the processor die separate from the cache memory. The list structure allows for access to a relatively small store of data to determine whether or not a cache entry needs to be written to the main memory. Choosing the format of the list structure allows one to make tradeoffs between area needed on a chip and the amount of efficiency in the cache flushing process.

List structure 44 is also shown in **Figure 4** in the context of cache 46, cache controller 48, request buffers 50, write-back buffers 52 and main memory 54, to illustrate how list structure

44 is used in a cache flush procedure. Cache controller 48 can generate evict requests and the set addresses for all the modified lines in cache 46 from list structure 44 alone, without having to directly access cache 46. The evict micro-operations and the set addresses will be stored in request buffers 50 which will issue the evict requests to cache 46. The data and address of the modified cache line will be latched in the write-back buffer. (*See* Paragraph 22).

6. ISSUES

A. Are claims 1-4, 7-12, 14, 16-27, and 29-30 rendered obvious over U.S Patent No. 5,860,105 to McDermott (Hereinafter "McDermott") in view of U.S Patent No. 6,490,657 to Masubuchi et al. (Hereinafter "Masubuchi") or Masabuchi in view of McDermott.?

B. Are claims 5-6 rendered obvious over McDermott in view of Masubuchi in further view of U.S Patent No. 6,058,456 to Arimilli (Hereinafter "Arimilli")?

C. Is claim 13 rendered obvious over McDermott in view of Masubuchi in further view of U.S Patent No. 6,460,122 to Otterness et al. (Hereinafter "Otterness")?

D. Are claims 15 and 28 rendered obvious over McDermott in view of Masubuchi in further view of U.S Patent No. 5,724,550 to Stevens (Hereinafter "Stevens")?

7. GROUPING OF CLAIMS

The claims are grouped as follows.

A. Claims 1-30.

The claims in these groups do not stand or fall together unless so indicated below in the argument.

8. ARGUMENT

A. Claims 1-4, 7-12, 14, 16-27, and 29-30 are not rendered obvious over McDermott in view of Masubuchi or Masabuchi in view of McDermott.

Independent claims 1, 16, 21, and 25 of the present invention describe a list structure to track the status of a plurality of cache entries. The list structure is located outside the cache and does not contain cache data or addresses. A query mechanism checks the list structure for the state of a cache entry. A cache flush mechanism flushes a cache entry and modifies the list structure to reflect a flushed state. Claims 2-4, 7-12, 14, 17-20, 22-24, 26-27, and 29-30 depend from claims 1, 16, 21, and 25.

Appellants respectfully submit that neither McDermott, Masubuchi, nor any combination thereof teaches or suggests a list structure for tracking a status of a plurality of cache entries, wherein the list structure is located outside a cache and does not contain cache data or addresses, as claimed in claims 1, 16, 21, and 25.

McDermott discloses an NDIRTY cache line lookahead technique used to expedite cache flush and export operations by providing a mechanism to avoid scanning at least some cache lines that do not contain dirty data (*See* Abstract). Masabuchi discloses a cache flush apparatus, attached to the system bus, that maintains the memory addresses held in dirty blocks within its own storage during the normal data processing. When a cache flush is required, the cache flush apparatus reads the memory addresses from the storage and issues bus commands. As a result, a dirty block becomes "shared" and it still holds the same data. For a non-dirty cache block, it remains unchanged (*See* Col. 2, Lines 49-57).

The Examiner has attempted to combine the two references to disclose this element. The Examiner states:

> Regarding Applicant's argument regarding the teachings of Masabuchi, it should be noted that Masabuchi is relied upon for teaching that which McDermott does not. McDermott does not teach the list structure located outside of a cache. Masabuchi is relied upon for teaching a list structure which is located outside of a cache. Hence, it is not clear why the Applicant is arguing that Masabuchi's list structure contains addresses, since Masabuchi was never relied upon for teaching that feature.

(Office Action, May 25, 2004, Page 12).

McDermott's list structure does not contain addresses because it is located inside the cache. Specifically, McDermott states:

> Referring to FIG. 2a, the NDIRTY cache line lookahead technique of the invention is implemented as part of the flush/export logic for the L1 cache 204.

(McDermott, Col. 6, Lines 57-59).

McDermott's list structure would be inoperable outside of the cache. Masabuchi does not make it possible for McDermott's list structure to operate outside of the cache. Masabuchi states:

> A cache flush device 30 and a memory controller 50 for a main memory 51, are connected to the system bus 40. The cache flush device 30 includes a system bus interface 31, an update address memory 32 (consisting of regions A_0 to A_{n-1}), an update address registering section 33, an update address removing section 34 and a flush executing section 35. The update address registering section 33 and the update address removing section 34 can be implemented as a single hardware module.
>
> The system bus interface 31 serves as an interface with the system bus 40.
>
> The update address memory 32 consists of n regions A_0 to A_{n-1} for storing addresses of data held in the dirty blocks of the cache memories 20. In this embodiment, if and only if a dirty block holds data of a certain address, the address is stored in a certain region of the update address memory 32. From now on, we call such a memory address stored in the update address memory 32 an update address.

(Masabuchi, Col. 11, Lines 5-23).

The Masabuchi list structure is entirely a list of addresses. The addresses are necessitated by the external nature of the list structure. Combining Masabuchi and McDermott would not produce a workable external list structure that does not contain cache data or addresses.

Therefore, a list structure for tracking a status of a plurality of cache entries, wherein the list structure is located outside a cache and does not contain cache data or addresses, as cited by claims 1, 16, 21, and 25.

The Examiner further states:

> Additionally, it should be noted that a reference does not teach away from a particular modification merely because it is silent to making the modification. Just because McDermott teaches a list structure within the cache does not necessarily infer that McDermott teaches away from having the list structure located outside of the cache.

(Office Action, May 25, 2004, Page 13).

McDermott does more than teach a list structure within the cache. McDermott states:

> An object of the invention is to store information in a cache array to reduce the time required for cache export and flush (export then invalidate) operations.

(McDermott, Col. 1, Lines 65-67).

McDermott teaches a list structure within the cache to speed up operation of the cache as an object of the invention. An external list would increase the time required for cache export and flush operations, which is counter to an express goal of the invention.

Appellants respectfully submit, therefore, that elements of claims 1, 16, 21, and 25 are neither shown nor suggested by the cited reference. Accordingly, the rejections of claims 1, 16, 21, and 25 under 35 U.S.C. §103(a) are in error, and Appellants respectfully request their reversal.

In summary, it has been demonstrated that the McDermott and Masabuchi references in combination do not suggest the recited claim combination. Obviousness under 35 U.S.C. §103(a) requires that the references disclose the claimed combination without reference to the application. Accordingly, a rejection of these claims under 35 U.S.C. §103(a) is improper. In

view of the above, Appellants respectfully submit that the rejection of claims 1-4, 7-12, 14, 16-27, and 29-30 should be reversed.

B. Claims 5-6 are not rendered obvious over McDermott in view of Masubuchi in further view of Arimilli.

Independent claim 1 of the present invention describes a list structure to track the status of a plurality of cache entries. The list structure is located outside the cache and does not contain cache data or addresses. A query mechanism checks the list structure for the state of a cache entry. A cache flush mechanism flushes a cache entry and modifies the list structure to reflect a flushed state. Claims 5-6 depend from claim 1.

Arimilli discloses allocating a cache used by a processor of a computer system between instructions and data (*See* Abstract).

Appellants respectfully submit that neither McDermott, Masabuchi, Arimilli, nor any combination thereof teaches or suggests a list structure for tracking a status of a plurality of cache entries, wherein the list structure is located outside a cache and does not contain cache data or addresses, as claimed in claim 1. The deficiencies in McDermott and Masabuchi are described above. Arimilli does not cure these deficiencies.

In summary, it has been demonstrated that the McDermott, Masabuchi, and Arimilli references in combination do not suggest the recited claim combination. Obviousness under 35 U.S.C. §103(a) requires that the references disclose the claimed combination without reference to the application. Accordingly, a rejection of these claims under 35 U.S.C. §103(a) is improper. In view of the above, Appellants respectfully submit that the rejection of claims 5-6 should be reversed.

C. Claim 13 is not rendered obvious over McDermott in view of Masubuchi in further view of Otterness.

Independent claim 1 of the present invention describes a list structure to track the status of a plurality of cache entries. The list structure is located outside the cache and does not contain cache data or addresses. A query mechanism checks the list structure for the state of a cache entry. A cache flush mechanism flushes a cache entry and modifies the list structure to reflect a flushed state. Claim 13 depends from claim 1.

Otterness discloses a multiple level cache structure and multiple level caching method that distributes I/O processing loads including caching operations between processors to provide higher performance I/O processing, especially in a server environment (*See* Abstract).

Appellants respectfully submit that neither McDermott, Masabuchi, Otterness, nor any combination thereof teaches or suggests a list structure for tracking a status of a plurality of cache entries, wherein the list structure is located outside a cache and does not contain cache data or addresses, as claimed in claim 1. The deficiencies in McDermott and Masabuchi are described above. Otterness does not cure these deficiencies.

In summary, it has been demonstrated that the McDermott, Masabuchi, and Otterness references in combination do not suggest the recited claim combination. Obviousness under 35 U.S.C. §103(a) requires that the references disclose the claimed combination without reference to the application. Accordingly, a rejection of these claims under 35 U.S.C. §103(a) is improper. In view of the above, Appellants respectfully submit that the rejection of claim 13 should be

reversed.

D. Claim 15 and 28 are not rendered obvious over McDermott in view of Masubuchi in further view of Stevens.

Independent claims 1 and 25 of the present invention describes a list structure to track the status of a plurality of cache entries. The list structure is located outside the cache and does not contain cache data or addresses. A query mechanism checks the list structure for the state of a cache entry. A cache flush mechanism flushes a cache entry and modifies the list structure to reflect a flushed state. Claims 15 and 26 depend from claims 1 and 25 respectively.

Stevens discloses responding to a microprocessor-generated write of a write-protected area of memory by invalidating a cache line corresponding to a write address in a microprocessor's internal cache by using a microprocessor address pin as a snoop invalidate signal during snoop cycles (*See* Abstract).

Appellants respectfully submit that neither McDermott, Masabuchi, Stevens, nor any combination thereof teaches or suggests a list structure for tracking a status of a plurality of cache entries, wherein the list structure is located outside a cache and does not contain cache data or addresses, as claimed in claims 1 and 26. The deficiencies in McDermott and Masabuchi are described above. Stevens does not cure these deficiencies.

In summary, it has been demonstrated that the McDermott, Masabuchi, and Stevens references in combination do not suggest the recited claim combination. Obviousness under 35 U.S.C. §103(a) requires that the references disclose the claimed combination without reference to the application. Accordingly, a rejection of these claims under 35 U.S.C. §103(a) is improper.

In view of the above, Appellants respectfully submit that the rejection of claims 15 and 26 should be reversed.

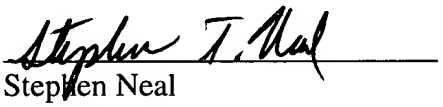
Appellants therefore respectfully request that the Board of Patent Appeals and Interferences reverse the Examiner's decision rejecting claims 1-30 and direct the Examiner to pass the case to issue.

The Examiner is hereby authorized to charge the appeal brief fee of **$340.00** and any additional fees which may be necessary for consideration of this paper to Kenyon & Kenyon Deposit Account No. **11-0600**.

Respectfully submitted,

KENYON & KENYON

Date: November 1, 2004

By: 

Stephen Neal
(Reg. No. 47,815)
Attorneys for Intel Corporation

KENYON & KENYON
333 West San Carlos St., Suite 600
San Jose, CA 95110

Telephone: (408) 975-7500
Facsimile: (408) 975-7501

APPENDIX

(Brief of Appellants Lokpraveen B. MOSUR et al.
U.S. Patent Application Serial No. 09/967,031)

CLAIMS ON APPEAL

1. An apparatus for cache flushing, comprising:

a list structure to track a status of a plurality of cache entries, wherein said list structure is located outside a cache and wherein said list structure does not contain cache data or addresses;

a query mechanism to check said list structure for the state of a cache entry; and

a cache flush mechanism, logically coupled to said list structure and the cache, to flush a cache entry and for modifying said list structure to reflect a flushed state.

2. An apparatus in accordance with claim 1, wherein:

said list structure comprises one bit per cache line.

3. An apparatus in accordance with claim 1, wherein:

said list structure comprises one bit per plurality of cache lines.

4. An apparatus in accordance with claim 1, wherein:

said list structure comprises one bit per cache way.

5. An apparatus in accordance with claim 1, further comprising:

one bit per a variable number of cache lines; and

wherein a logical arrangement of said list structure conforms to said variable number.

6. An apparatus in accordance with claim 5, wherein:

said variable number is set by an operating system.

7. An apparatus in accordance with claim 1, wherein:

a logical arrangement of said list structure matches an architecture of a cache.

8. An apparatus in accordance with claim 1, wherein:

said cache flush mechanism modifies a cache state responsive to results of a query of the said list structure.

9. An apparatus in accordance with claim 8, wherein:

said cache flush mechanism is logically coupled to a higher level cache for writing back modified data.

10. An apparatus in accordance with claim 8, wherein:

said cache flush mechanism based on the said list structure is logically coupled to a higher level cache for evicting modified data.

11. An apparatus in accordance with claim 8, wherein:

said cache flush mechanism is logically coupled to a main memory for writing back modified data.

12. An apparatus in accordance with claim 8, wherein:

said cache flush mechanism is logically coupled to a main memory for evicting modified data.

13. An apparatus in accordance with claim 1, wherein:

said list structure is located in random access memory (RAM).

14. An apparatus in accordance with claim 1, wherein:

said list structure is located on a die.

15. An apparatus in accordance with claim 1, further comprising:

a snoop command interpreter to check said list structure in response to a snoop command.

16. In a computer system with a cache memory, an apparatus for flushing the cache, comprising:

a list structure to record modifications to a plurality of cache entries wherein, wherein said list structure is located outside a cache and said list structures does not contain cache data or addresses;

a cache controller to query said list structure for modifications to said plurality of cache entries and generate a list of cache write-back instructions; and

wherein said cache controller is to invalidate said plurality of cache entries corresponding to said list of cache write-back instructions.

17. An apparatus in accordance with claim 16, wherein:

said list structure is a full list.

18. An apparatus in accordance with claim 16, wherein:

said list structure is a partial list.

19. An apparatus in accordance with claim 17, wherein:

said full list comprises one entry per cache line.

20. An apparatus in accordance with claim 18, wherein:

said partial list comprises one entry per plurality of cache lines.

21. In a multiprocessor computer system with a plurality of processors and cache memory, an apparatus for cache flushing, comprising:

a list structure to track a status of a plurality of cache entries, wherein said list structure is located outside a cache and wherein said list structure does not contain cache data or addresses;

a processor identification within said list structure to link each of said plurality of cache entries to one of the plurality of processors;

a query mechanism to check said list structure for a state of a cache entry identified with a processor;

a cache flush mechanism to flush a cache entry linked to an identified processor and to modify said list structure to reflect a flushed status.

22. An apparatus in accordance with claim 21, wherein:

said list structure contains at least one bit for each cache line.

23. An apparatus in accordance with claim 21, wherein:

said list structure contains at least one bit for each of a plurality of cache lines.

24. An apparatus in accordance with claim 21, wherein:

said list structure is located on a die with at least one of the plurality of processors.

25. A method of flushing a cache, comprising:

creating a table of cache entries separate from the cache and without the cache data or addresses;

tracking modified cache entries in said table; and

generating a write-back command from said table in response to a cache flush event.

26. A method in accordance with claim 25, further comprising:

generating an invalidate command in response to a cache flush event.

27. A method in accordance with claim 25, further comprising:

repeating the method for each level of cache.

28. A method in accordance with claim 25, further comprising:

querying said table in response to a snoop command.

29. A method in accordance with claim 25, further comprising writing-back modified cache entries to memory.

30. A method in accordance with claim 25, further comprising: writing-back modified cache entries to a high level cache.